

## **REMARKS:**

### **Claim Rejections 35 USC §112**

The Examiner has rejected claim 1 under 35 USC §112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which Applicant regards as his invention.

Applicant has amended claims 1 and 22 to change the phrase “cache components” to “cache objects”. This change further clarifies claims 1 and 22 and is supported by the specification on page 5, line 28, page 6, lines 1-5, and page 11, line 24, for example. No new matter has been added.

### **Claim Rejections 35 USC §101**

The Examiner has rejected claims 1 and 22 under 35 USC §101 because the Examiner states the claimed invention is directed to non-statutory subject matter.

Claim 1 and claim 22 have been amended to reflect that the method and structure are within a computer having a cacheable memory. This modification is clearly supported in the specification on page 2, lines 11-18. No new matter has been added.

### **Claim Rejections 35 USC §103**

The Examiner has rejected Claims 1-6, 12-14, 16, 19, 20, 22-29, and 33 under 35 U.S.C. 103(a) as being unpatentable over US Pat. 5,943,497 (Bohler et al, hereinafter called Bohler) in view of US Pat. 6,438,743 (Boehm et al, hereinafter called Boehm). Applicant respectfully traverses this rejection of the claims.

Prior to addressing the claim rejection, applicant notes that a reading of Bohler teaches (summary) an object-oriented mechanism that allows new configuration

data to replace existing configuration data within an existing object oriented program (OOP). This mechanism allows class substitution or redefinition in an OOP.

A reading of Boehm teaches (summary), "A method and apparatus for building a software system in a networked software development environment, utilizing existing software version control and build tools such as RCS and MAKE. At individual workstations, a cache link structure generated from a user-created build list is provided to the software building program, which then builds the desired software system using links to cached files. The present invention thus minimizes the amount of computing resources required to build software programs by eliminating the need to store multiple local copies of building block software files, and to rebuild object files that may be unchanged from prior builds. A method for maintaining and updating network caches to maximize the efficiency of cache link creation is also disclosed."

**With regard to claim 1**, Applicant strenuously objects to the Examiner's characterization that the combination of the Bohler and Boehm references obviates applicant's claim 1. As will be shown below, the combination of Bohler and Boehm does not teach or suggest the recitations in total of rejected claim 1.

The Examiner has stated that Bohler teaches on col 2, lines 24-27 "a method for run-time configurable caching of component factories" which has been amended in claim 1 to read "a method for run-time configurable caching of factory objects". However, a reading of Bohler shows that there is no teaching, suggestion or anticipation of run-time configurable caching of factory objects (or component factories). In fact, the Applicant has found no mention in Bohler of run-time configurable caching of factory objects, component factories, or even run-time configurable caching. Bohler does teach the ability to replace configuration data while an object oriented application is running, thereby allowing a user to create an object while the application is running (summary, col. 6, lines 52-56). But the Applicant notes that methods and structures for creating an object during run-time has not been related by Bohler to methods and apparatus for creating and

manipulating cache memory during run-time. And, significantly, Examiner has not stated with particularity where such teaching occurs.

The Examiner is kindly directed to col. 6, lines 52-56, of Bohler which states, "The disclosed embodiments avoid the difficulties with working with existing OO programs (e.g., frameworks) by allowing data administrator control over the modification of configuration data without having to change the whole program." So, this reading of Bohler gives a user external control over an OO program without having to re-compile changes to the program which is not related to Applicant's claim 1.

Furthermore, a reading of col. 6, lines 52-56 actually **teaches away** from Applicant's claim 1, since Bohler teaches how to "avoid the difficulties with working with existing OO programs (e.g., frameworks)". Since Factories are often part of frameworks in the art, and the invention of Bohler is directed to a mechanism to avoid frameworks, Applicant's claim 1 should not be so limited.

In summary, the Applicant asserts that Bohler does not teach, suggest, anticipate or otherwise obviate an ability to configure or manipulate a cache of factory objects of an application while the application is running. And, significantly, Examiner has not stated with particularity where such teaching occurs. Because the combination of the Bohler and Boehm references does not teach every element of claim 1 as currently amended, the rejections of claim 1 are unsupported by the art and should be withdrawn. Reconsideration and allowance of claim 1 is hereby requested at the Examiner's earliest convenience.

Applicant also notes that the fourth element of Applicant's claim 1, "assigning one or more cache objects of the plurality of cache objects to one or more cacheable factory objects of the plurality of cacheable factory objects", is not taught, suggested, anticipated or otherwise obviated by Bohler. There is no teaching in Bohler of a plurality of cache objects, a plurality of cacheable factory objects or any kind of assignment between these two pluralities. Examiner has stated that col. 6, lines 46-50 obviate Applicant's fourth element of claim 1. **However, a reading of Bohler only indicates the use of a cache (not a cache object) to access configuration data objects.** There is no teaching, suggestion, or obviation in Bohler of the use of

cache objects. In fact, a reading of Col 6, lines 46-50 indicate that there is a single cache which is accessed using the namingService object. *Since there is no teaching of the use of cache objects or cacheable factory objects, there certainly can not be any teaching, suggestion or obviation of any kind of mapping between cache objects and cacheable factory objects.* Therefore, Applicant asserts that Bohler does not teach, suggest, anticipate or otherwise obviate an assignment between cache objects and cacheable factory objects. And, significantly, Examiner has not stated with particularity where such teaching occurs. Because the Bohler reference does not teach every element of claim 1 as currently amended, the rejections of claim 1 are unsupported by the art and should be withdrawn. Reconsideration and allowance of claim 1 is hereby requested at the Examiner's earliest convenience.

While Applicant heartily agrees that Bohler does not teach elements 2, 3 or 5 of Applicant's claim 1, Applicant strenuously objects to the Examiner's assertion that the Boehm reference teaches any elements of Applicant's claim 1. With reference to the Summary of Boehm and Figures 2 and 4, Boehm teaches a method and apparatus for creating a software application and has **no relevance** to how said software application manages its cache using objects or otherwise. **In fact, the object files referred to in the Boehm reference are only the files produced using a program configuration tool that compiles source files to create object files. A reading of Boehm finds no mention of object oriented programming principles whatsoever. The object files of Boehm are not objects in the OO sense.** And, significantly, Examiner has not stated with particularity where such teaching occurs.

Applicant therefore asserts that reliance upon the Boehm reference is defective, and thus the combination of the Bohler reference with the Boehm reference is flawed. Applicant asserts that Boehm is not analogous art, since Boehm is outside the field of the Applicant's invention, having no teaching on object oriented applications. Because the combination of the Bohler reference with the Boehm reference does not teach every element of claim 1 as currently amended, the rejections of claim 1 are unsupported by the art and should be withdrawn. Reconsideration and allowance of claim 1 is hereby requested at the Examiner's earliest convenience.

**With regard to claims 2-21**, Applicant submits that these claims are patentable because these claims depend from claim 1 which has been shown to be patentable. Although additional arguments could be made for the patentability of claims 2-21, such arguments are believed unnecessary in view of the above discussion and the changes to claims 2-21. The undersigned wishes to make it clear that not making such arguments at this time should not be construed as a concession or admission to any statement in the Office Action.

Applicant further asserts that the Examiner's use of the combination of Bohler and Boehm as a reference for rejection of claims 2, 5, 13, 20, 22, 23, 25, 27, 28, 29, and 33 is improper because as mentioned in the response to the claim 1 rejection, reliance upon the Boehm reference in combination with the Bohler as a grounds of rejection is defective. Therefore the rejections of claims 2, 5, 13, 20, 22, 23, 25, 27, 28, 29, and 33 are defective and should be withdrawn. Applicant further reiterates that the use of the Boehm reference is defective for the reasons stated in the response to the claim 1 rejection. With reference to the Summary of Boehm and Figures 2 and 4, Boehm teaches a method and apparatus for creating a software application and has **no relevance** to how said software application manages its cache using objects or otherwise. **In fact the object files referred to in the Boehm reference are only the files produced using a program configuration tool that compiles source files to create object files. A reading of Boehm finds no mention of object oriented programming principles whatsoever.** And, significantly, Examiner has not stated with particularity where such teaching occurs. Because the combination of the Bohler reference with the Boehm reference does not teach every element of claims 2, 5, 13, 20, 22, 23, 25, 27, 28, 29, and 33, the rejections of claims 2, 5, 13, 20, 22, 23, 25, 27, 28, 29, and 33 are unsupported by the art and should be withdrawn. Reconsideration and allowance of claim claims 2, 5, 13, 20, 22, 23, 25, 27, 28, 29, and 33 are hereby requested at the Examiner's earliest convenience.

**With regard to claim 6**, Applicant agrees that Bohler does not disclose the element of the claim. However, claim 6 depends from claim 1 which has been shown to be

patentable. Since no other reason for a rejection of claim 6 has been given, reconsideration and allowance of claim 6 is hereby requested at the examiner's earliest convenience.

**With regard to claims 22 and 27,** Applicant strenuously objects to the Examiner's characterization that the combination of the Bohler and Boehm references obviate Applicant's claim 22 and claim 27. As will be shown below, the combination of Bohler and Boehm does not cover the entirety of any of the rejected claims 22 or 27.

The Examiner has stated that Bohler teaches on col 9, lines 45-47 "one or more cacheable factory objects residing in a memory of the general purpose computer, derived from the one or more factory objects". However, a reading of Bohler shows that there is no teaching, suggestion or obviation of cacheable factory objects derived from one or more factory objects. In fact, the Applicant has found no mention in Bohler of run-time configurable caching of factory objects, cacheable factory objects, or even run-time configurable caching. And, significantly, Examiner has not stated with particularity where such teaching occurs.

The Examiner is kindly directed to col. 6, lines 52-56, which state "The disclosed embodiments avoid the difficulties with working with existing OO programs (e.g., frameworks) by allowing data administrator control over the modification of configuration data without having to change the whole program." So, this reading of Bohler gives a user external control over an OO program without having to re-compile changes to the program. Furthermore, a reading of col. 6, lines 52-56 actually **teaches away** from Applicant's claim 22, since Bohler teaches how to "avoid the difficulties with working with existing OO programs (e.g., frameworks)". Since Factories can be part of frameworks, Applicant's claim 22 should not be so limited.

In summary, the Applicant asserts that Bohler does not teach, suggest, anticipate or otherwise obviate an ability to have cacheable factory objects derived from one or more factory objects while the application is running. And, significantly, Examiner



has not stated with particularity where such teaching occurs. Because the Bohler reference does not teach every element of claim 22 as currently amended, the rejections of claim 22 are unsupported by the art and should be withdrawn. Reconsideration and allowance of claim 22 is hereby requested at the Examiner's earliest convenience.

While Applicant heartily agrees that Bohler does not teach the third element of Applicant's claim 22, Applicant strenuously objects to the Examiner's assertion that the Boehm reference teaches the third element of Applicant's claim 22. With reference to the Summary of Boehm and Figures 2 and 4, Boehm teaches a method and apparatus for creating a software application and has **no relevance** to how said software application manages its cache using objects or otherwise. **In fact the object files referred to in the Boehm reference are only the files produced using a program configuration tool that compiles source files to create object files. A reading of Boehm finds no mention of object oriented programming principles whatsoever. The object files of Boehm are not objects in the OO sense.** And, significantly, Examiner has not stated with particularity where such teaching occurs.

Applicant therefore asserts that the Boehm reference is defective, and thus the combination of the Bohler reference with the Boehm reference is flawed. Because the combination of the Bohler reference with the Boehm reference does not teach every element of claim 22 as currently amended, the rejections of claim 22 is unsupported by the art and should be withdrawn. Reconsideration and allowance of claim 22 is hereby requested at the Examiner's earliest convenience.

Applicant further asserts that the Examiner's use of the combination of Bohler and Boehm as a reference for rejection of claims 23, 25, 28, 29, and 33 is improper because as mentioned in the response to the claim 1 rejection, the Boehm reference is defective. Therefore the rejections of claims 23, 25, 28, 29, and 33 are defective and should be withdrawn. Applicant further reiterates that the use of the Boehm reference is defective for the reasons stated in the response to the claim 1 rejection. With reference to the Summary of Boehm and Figures 2 and 4, Boehm teaches a

method and apparatus for creating a software application and has **no relevance** to how said software application manages its cache using objects or otherwise. **In fact the object files referred to in the Boehm reference are only the files produced using a program configuration tool that compiles source files to create object files. A reading of Boehm finds no mention of object oriented programming principles whatsoever.** And, significantly, Examiner has not stated with particularity where such teaching occurs. Because the combination of the Bohler reference with the Boehm reference does not teach every element of claims 23, 25, 28, 29, and 33, the rejections of claims 23, 25, 28, 29, and 33 are unsupported by the art and should be withdrawn. Reconsideration and allowance of claim claims 23, 25, 28, 29, and 33 are hereby requested at the Examiner's earliest convenience.

**With regard to claims 23-33,** Applicant submits that these claims are patentable because these claims depend from claim 22 which has been shown to be patentable. Although additional arguments could be made for the patentability of claims 23-33, such arguments are believed unnecessary in view of the above discussion and the changes to claims 23-33. The undersigned wishes to make it clear that not making such arguments at this time should not be construed as a concession or admission to any statement in the Office Action.

The Examiner has rejected Claims 7-11, 15, 17, 18, and 30-32 under 35 U.S.C. 103(a) as being unpatentable over US Pat. 5,943,497 (Bohler) in view of US Pat. 6,446,188 (Henderson et al., hereinafter called Henderson). Applicant respectfully traverses this rejection of the claims. Applicant would like to first note that claims 7-11, 15, 17, 18 depend from claim 1, which has been shown to be patentable and claims 30-32 depend from claim 22 which has also been shown to be patentable over the cited references. Thus, the rejection of claims 7-11, 15, 17, 18, and 30-32 are improper and should be withdrawn.

Applicant would like to further assert that the use of Henderson is improper because Henderson, as indicated in the Summary, provides "A system for dynamic memory management maps a sparsely populated virtual address space of memory objects to a more densely populated physical address space of fixed size memory elements for

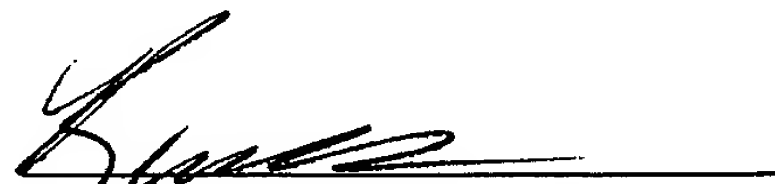


use by a host processor." The memory objects mentioned therein are not objects in the object-oriented sense, but are objects within the virtual address space. Henderson does not teach, suggest, or obviate the use of object oriented programming or design and so Henderson is certainly not applicable to the Applicant's claims 7-11, 15, 17, 18, and 30-32. Applicant therefore asserts that Henderson is not analogous art, since Henderson is outside the field of the Applicant's invention, having nothing to do with object oriented applications. Thus, the rejection of claims 7-11, 15, 17, 18, and 30-32 are improper and should be withdrawn. Reconsideration and allowance of claims 7-11, 15, 17, 18, and 30-32 are hereby requested at the Examiner's earliest convenience.

Although additional arguments could be made for the patentability of claims 1-33, such arguments are believed unnecessary in view of the above discussion and the changes to claims 1, and 22. The undersigned wishes to make it clear that not making such arguments at this time should not be construed as a concession or admission to any statement in the Office Action.

Please contact the undersigned if there are any questions regarding this response or application.

Respectfully submitted,



Renee' Michelle Larson  
Registration No. 36,193  
Larson and Associates, P.C.  
221 East Church Street  
Frederick, MD 21701  
Phone (301) 668-3073  
Fax (301) 668-3074

Dated: September 20, 2004